AMENDMENTS TO THE CLAIMS

Please amend the claims as set forth below in marked-up form. This listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently Amended) A computing machine having a computing architecture, comprising:

a base operating system (OS) installed to the computing machine, the base OS having a base OS file system and a base OS registry;

at least one virtual OS environment within the base OS, the virtual OS environment having a virtual OS file system and a virtual OS registry which are independent of the base OS file system and the base OS registry[[;]], wherein ~~injecting~~ a dynamic link library (DLL) is injected into an application running under the virtual OS environment[[;]], the DLL being programmed to

~~redirecting, via the DLL, attempts of the application to access the base OS file system and the base OS registry to the virtual OS file system and the virtual OS registry~~

determine that the application should be run under the virtual OS environment instead of the base OS;

call a function of the injected DLL rather than calling an API of the base OS;

modify at least one parameter from a calling function of the application to direct the at least one parameter to a location of the virtual OS environment;

after modifying the at least one parameter, call the API of the base OS with the at least one modified parameter;

receive information from the API of the base OS;

modify the information from the API of the base OS to convert the information back from a virtual OS environment location;

return the information to the calling function of the application.

2. (Canceled)

3. (Previously Presented) The computing machine of claim 1, wherein the application running under the virtual OS environment shares one or more of the following with the base OS:

working information,
user login rights,
services,
hardware information, and
clipboard information.

4. (Previously Presented) The computing machine of claim 1, further including multiple virtual OS environments within the base OS, and wherein a change made in one of the virtual OS environments does not affect the base OS or any of the other virtual OS environments.

5. (Previously Presented) The computing machine of claim 1, wherein each virtual OS environment contains a group of installed applications that run independently of one another.

6. (Previously Presented) The computing machine of claim 1, further including one or more applications running under the base OS and each virtual OS environment, and wherein all of the applications run on a single OS desktop.

7. (Previously Presented) The computing machine of claim 1, wherein a change made to configuration information with respect to the virtual OS environment does not change configuration information associated with the base OS.

8. (Currently Amended) A method of configuring a computer with a base operating system (OS) having a base OS file system and registry, the method comprising the steps of:

creating at least one virtual OS environment under the base OS, each virtual OS environment having a virtual OS file system and registry which are independent of the base OS file system and registry;

~~configuring the computer such that attempts to access the base OS file system and registry by at least one application running under the virtual OS environment are redirected to the virtual OS environment file system and registry;~~

injecting a DLL into every application that is executed under the virtual OS environment;

determining that at least one application should be run under the virtual OS environment instead of the base OS, wherein the DLL performs the determining;

calling a function of the injected DLL rather than calling an API of the base OS, wherein the DLL performs the calling;

modifying at least one parameter from a calling function of the application to direct the at least one parameter to a location of the virtual OS environment, wherein the DLL performs the modifying.

after modifying the at least one parameter, calling the API of the base OS with the at least one modified parameter, wherein the DLL performs the calling;

receiving information from the API of the base OS, wherein the DLL performs the receiving;

modifying the information from the API of the base OS to convert the information back from a virtual OS environment location, wherein the DLL performs the modifying of the information;

returning the information to the calling function of the application, wherein the DLL performs the returning.

9. (Canceled)

10. (Previously Presented) The method of claim 8, further including the step of altering one or more application programming interfaces (APIs) that access the base OS file system and registry directly and indirectly so as to redirect these accesses into the appropriate virtual OS file system and registry.

11. (Canceled)

12. (Previously Presented) The method of claim 8, further including the step of creating a copy of the base OS file system and registry in the virtual OS environment file system and registry.

13. (Previously Presented) The method of claim 12, wherein the application running under the virtual OS environment is executed using the copy in the virtual OS environment file system and registry.

14. (Previously Presented) The method of claim 8, further including setting a predetermined directory such that an application running under the predetermined directory will be redirected to the virtual OS environment based on the location of the application being under the predetermined directory.

15. (Previously Presented) The method of claim 14, wherein the predetermined directory is a CD/DVD drive in the base OS file system.

16. (Currently Amended) A method of configuring a computer with a base operating system (OS) having a base OS file system and a base OS registry, the method comprising the steps of:

creating a virtual OS environment within the base OS, the virtual OS environment having at least one of:

a virtual OS file system that is independent of the base OS file system;

a virtual OS registry that is independent of the base OS registry;

injecting a dynamic link library (DLL) into an application;

redirecting, via the DLL, an attempt by the application to access the base OS to at least one of:

the virtual OS file system of the at least one virtual OS environment;

the virtual OS registry of the at least one virtual OS environment;

wherein the redirecting comprises:

determining that the application should be run under the virtual OS environment instead of the base OS, wherein the DLL performs the determining;

calling a function of the injected DLL rather than calling an API of the base OS, wherein the DLL performs the calling;

modifying at least one parameter from a calling function of the application to direct the at least one parameter to a location of the virtual OS environment, wherein the DLL performs the modifying.

after modifying the at least one parameter, calling the API of the base OS with the at least one modified parameter, wherein the DLL performs the calling;

receiving information from the API of the base OS, wherein the DLL performs the receiving;

modifying the information from the API of the base OS to convert the information back from a virtual OS environment location, wherein the DLL performs the modifying of the information;

returning the information to the calling function of the application, wherein the DLL performs the returning.

17. (Currently Amended) The method of claim 16, further comprising:

~~determining that the application should be run under the virtual OS environment instead of the base OS, wherein the DLL performs the determining;~~

scanning a function import table of the application, wherein redirecting the attempt by the application comprises redirecting, via the DLL, file system and registry calls from the application to functions within the injected DLL, wherein the DLL performs the scanning.

18. (Cancelled)

19. (Cancelled)

20. (Previously Presented) The method of claim 16, wherein:

the attempt to access the base OS comprises an attempt to install at least one file in the base OS file system.

21. (Previously Presented) The method of claim 20, wherein:

the attempt to install the at least one file in the base OS file system comprises an attempt to reboot the computer.

22. (Previously Presented) The method of claim 20, further comprising:

shutting down the application running under the at least one virtual OS environment;

after shutting down the application, installing the at least one file in the virtual OS file system of the at least one virtual OS environment;

after installing the at least one file, executing the application under the at least one virtual OS environment.

23. (Previously Presented) The method of claim 22, wherein:

shutting down the application running under the at least one virtual OS environment and installing the at least one file in the virtual OS file system of the at least one virtual OS environment are conducted while the base OS is running.